

مدخل إلى

Physical Computing

اعداد وتقديم

أحمد جمال خليفة عبدالعال

بسم الله الرحمن الرحيم
السلام عليكم ورحمة الله وبركاته

مقدمة بسيطة عن عالم ال physical computing .

- **اعداد وتقديم : أحمد جمال خليفة .**

- **تم نشر سلسلة الدروس في منتدى فيجوال بيسك للعرب خلال مطلع عام 2008 .**

- **المادة العلمية اصلاً قدمت بواسطة FUTEX Group في مشروع مادة Seminar في الفرقة الثالثة – كلية الحاسبات والمعلومات – حلوان – الدكومنتيشن الخاصة بالعرض باللغة الانجليزية مرفقة بالكتاب .**

- **لاي استفسارات :**
A-Gamal@windowslive.com

الفهرس

الدرس الأول : تمثيل البيانات في الكومبيوتر والدوائر الرقمية

هذا الدرس موجه للمبتدئين على عجالة من أجل فهم كيفية عمل الكومبيوتر...
لن أطيل ، ولمن يريد الاستزادة يمكنه البحث على الإنترنت ، ابتداء من الدروس القادمة سوف افترض مستوى معين للشرح.

بداية ، كيف يعمل الكومبيوتر ؟
الكومبيوتر عبارة عن جهاز كهربائي لا يفهم سوى لغة الكهرباء ، وما دمنا نتحدث من هذا المنطلق فليس للكهرباء أي حالة سواء (مفتوح ، مغلق) .

وهو ما نعبر عنه في حالة الكومبيوتر ب 0 و 1 .

وكيف يتعامل الحاسب مع هذه البيانات ؟

هناك ثلاث معاملات منطقية رئيسية تتم من خلالها جميع عمليات الكومبيوتر ، هم:

Not

```
حيث أن Not 1 = 0 و Not 0 = 1
```

AND
حيث أنه لا بد للطرفين أن يكونا = 1 ليكون الناتج 1 ، فمثلاً:
كود:

```
1 And 1 = 1 , 1 And 0 = 0 ... etc
```

OR
حيث لا بد لأحد الطرفين أن يكون 1 ليكون الناتج 1 فمثلاً:

```
1 Or 1 = 1 , 1 Or 0 = 1 , 0 Or 0 = 0
```

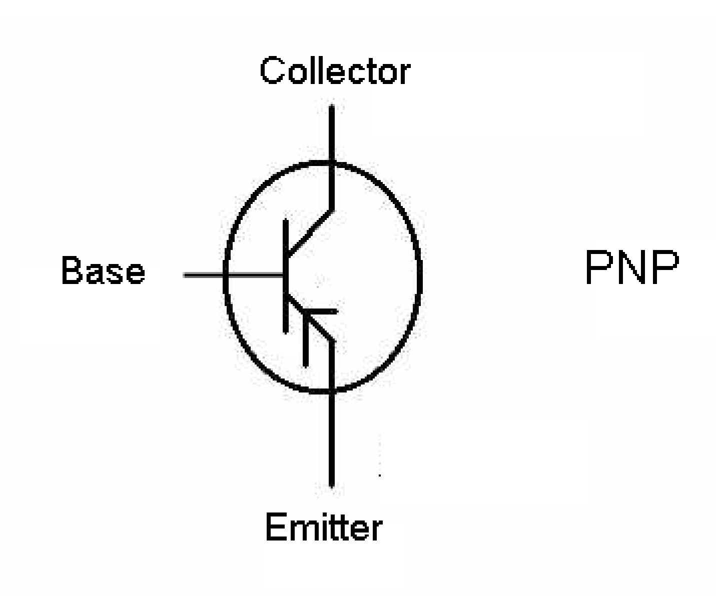
ومن خلال هذه العمليات نستطيع القيام بما نريد من خلال الحاسب.

الترانزستور :

ربما من خلال الشرح السابق تكون قد سألت نفسك بعض الأسئلة ، ما دمنا قلنا في البداية ان الكومبيوتر لا يتعامل سوى مع الكهرباء ، فكيف يستطيع القيام بمثل هذه العمليات.

الجواب أنه وبعد جيل من المحاولات المختلفة (مثل الصمامات المفرغة) ظهرت لنا الترانزستورات وفي أبسط أحوالها فإن القيمة الكهربائية تتجاوز من نقطة 1 إلى 3 إذا كانت نقطة 2 = 1 ، فيما عدا ذلك لا يمكنها التجاوز.

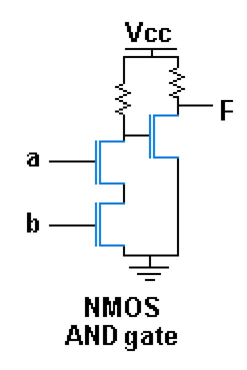
وهذا تخطيط لترانزستور مثلاً:



وللمزيد حول الترانزستور تستطيع البدء من هنا:

http://ar.wikipedia.org/wiki/%D8%AA%...%D9%88%D8%B1

وماذا بعد ، كيف استطيع الاستفادة من الترانزستور في هذه العمليات ؟
لن يفيدك كمبتدئ التعمق كثيراً في هذه النقطة ، عموماً هذه طريقة لكيفية استخدام الترانزستورات في بناء AND .



وماذا بعد:

من خلال هذه الدوال الرئيسية تم انشاء العديد من الدوال الأخرىمثل XOR,NAND ..الخ ، لننتقل بعدها إلى ما يسمى بالدوائر المتكاملة.

الدوائر المتكاملة.

لاحقاً تم تجميع هذه العمليات في دوائر مغلقة لتسهيل التعامل ، وجرى تجميع أكثر من واحدة في نفس المكان ، لا تنس أن كل دائرة تحتاج إلى مصدر VCC كهربائي حسب الدائرة اضافة إلى مصدر أرضي. Ground

نقطة أخرى : التخزين.

لتخزين البيانات نحتاج إلى نظام تمثيل ، فإذا كان نظامنا لا يتعمل سوى مع ارقام من واحد إلى سبعة فنحن نحتاج إلى ثلاث خانات فقط للتمثيل:

0:000
1:001
2:010
3:011
4:100
5:101
6:110
7:111

وهكذا ، وفي نظامنا الحالي نستخدم كمية كبيرة جداً الرموز والأرقام والحروف ، لذا نستخدم نظام ترميز مناسب.

نقطة أخيرة:

كما لاحظت فمرغماً لم أستطع التوسع في الشرح في هذه النقاط لإني لست مؤهل للشرح بهذه الطريقة في هذا المجال ، لذا سأفترض من المبتدئين أنهم سيطالعون الكتب في آخر هذا الدرس ، وابتداء من الدرس القادم سأتحدث على افتراض انهم قد فهموا ما شرحته في هذا الدرس - بدون تفاصيل متعمقة بالطبع. -

الكتب:

http://www.nekhely.com/Main/Kits/KITS-electrobook.htm

ستجد هنا المكتبة كاملة وهي مفيدة للمحترفين ، وكمبتدء لن أطلب الغوص فيها كثيراً بل نبذة عن كل شيء في حدود دقيقة أو اثنتين كافية جداً لتوصيل الهدف الذي اريد الوصول إليه ، ولمعرفة هل استطعت الفهم بدرجة كافية لمتابعة الدورة اسأل نفسك هذه الأسئلة وأجب عليها:

1- ماذا يعني - IC دوائر متكاملة - ؟
2- كما هي فكرة عمل الكومبيوتر (البروسيسور) ؟
3- ما هي المصادر الكهربائية المطلوبة للدوائر VCC,Ground ؟
4- ما هو الفرق بين المصدر الكهربائي للعمليات والمصدر الكهربائي في السؤال السابق ؟
5- هل استطعت تخيل ما يمكن أن نقوم بعمله من خلال الصفر والواحد ؟

فقط لن أطلب منك المزيد ، بعد الانتهاء ابدء مباشرة بمتابعة الدروس التالية .

الدرس الثاني : نظم الأرقام

كما ذكرنا في الدرس السابق فإننا بحاجة إلى نظام أرقام يفهمه الكومبيوتر ، وكان أن اخترنا النظام الثنائي binary بسبب طبيعته المحدودة.

وكما افترضنا في الدرس السابق الأرقام من 1 إلى 7 وطريقة تمثيلها سوف نتعلم هنا طريقة التحويل من رقم عشري إلى رقم ثنائي والعكس.

مثلاً : الرقم الثنائي : 00010111
طريقة العمل ، سوف نبدأ بالأرقام من اليمين إلى اليسار ، ونضرب كل رقم في 2^مكانه ، بمعنى:
الرقم الأول * 2^0 = 1*2^0 = 1*1 = 1
الرقم الثاني * 2^1 = 1*2^1 = 22*1 =
الرقم الثالث * 2^2 = 1*2^2 = 4*1 = 4
الرقم الرابع * 2^3 = 1*2^3 = 08*0 =
الرقم الخامس * 2^4 = 1*2^4 = 16*1 = 16
الرقم السادس * 2^5 = 1*2^5 = 032*0 =
الرقم السابع * 2^6 = 1*2^6 = 64*0 = 0
الرقم الثامن * 2^7 = 1*2^07 = 0*128 =

لا تنس أن ^ تعني (أس - to the power of) .

والآن نقوم بجمع الأرقام الناتجة : 1+2+4+16 = 23 وهي ما يرمز له بـ 00010111 في نظام يتم تمثيل الأرقام فيه في 8 بت (8 خانات.)

العملية العكسية ، الرقم 18 مثلاً (في نظام ترقيم يتم تمثيل الأرقام فيه في 8 بت.)
9 = 2 / 18والباقي (صفر).
4 = 2 / 9والباقي (واحد).
2 = 2 / 4والباقي (صفر).
1 = 2 / 2والباقي (صفر).
0 = 2 / 1والباقي (واحد).

والآن سوف نقوم برص الأرقام منذ البداية ، ونكمل الباقي حتى 8 خانات بأصفار ، ولذا يكون الناتج. 00010010

ولكن ماذا عن الرموز والحروف ؟

هناك ما يعرف بنظم الترميز مثل ASCII آسكي ، و Unicode يونيكود وغيرها ، ولنر مثلاً كيف يقوم نظام آسكي بالتعرف على الحروف والرموز.

يفترض نظام آسكي American Standard Code for Information Interchange سعة 8 بت لتمثيل كل خانة.

وبهذ الطريقة يمكننا تخزين المعلومات عن 2^8 رمز أو رقم أو حرف وهو ما يساوي 256.

***لا تنس أن الرقم 14 مثلاً يتم تخزين (4) كرقم و (1) كرقم آخر.**

في آسكي يتم حجز الأرقام من 0-31 اضافة للرقم 127 للرموز الخاصة مثل Delete وغيرها.
والأرقام من 32-47 من أجل الرموز مثل ! +
والأرقام من 48-57 يتم فيها تمثيل الأرقام من 1 - 9.
من 58-64 : رموز أخرى مثل ؟
من 65-90 الحروف الإنجليزية. Capital
من 91-96 : رموز مرة أخرى مثل [
من 97-122 : الحروف الإنجليزية Small
من 123-127 : رموز أخرى.

وتبقى لدينا 127 أخرى يتم تشكيلها حسب الحاجة حيث تجد مثلاً في حالتنا الحروف العربية وخلافه.

وللاستزادة يمكنك الإطلاع على هذا الرابط:
http://en.wikipedia.org/wiki/ASCII

الدرس الثالث -المنافذ وتركيبها وطريقة تعامل النظام معها

في هذا الدرس سوف نلقي نظرة سريعة على عالم منافذ الكومبيوتر ونتعرف عليها باختصار قبل ان نتوسع في الدرس القادم في استخدام منفذ الطابعة وكيفية برمجته والتعامل معه.

باختصار فإن منافذ الكومبيوتر على اختلاف انواعها ما هي إلا وسيلة لاخراج البيانات خارج نطاق الكومبيوتر أو ادخالها إليه عبر أجهزة مختلفة أو العكس ، ومن أبسط الأمثلة على ذلك لوحة المفاتيح مثلاً والتي تستخدم لادخال البيانات عبر منفذ USB أوPS2 ،وفي المقابل الطابعة والتي تستخدم ايضاً Parallel Portأو USB .

وببساطة فإن جميع هذه الأجهزة تقوم باستقبال 0 و 1 أو ارساله إلى الكومبيوتر إلى برنامج مختص بذلك يقوم بفهمها وتحليل المطلوب ، ففي حالة الكيبورد وبعد استلام اشارة كهربائية يقوم Driver خاص بترجمة هذه الاشارة إلى حرف أو رمز ، وفي العملية العكسية في الطابعة فإن الاشارات الكهربائية التي تتسلمها الطابعة يتم تحويلها إلى مجموعة من الخطوط والحروف والألوان .

المنفذ التسلسلي Serial Port :



صمم هذا المنفذ قديماً من أجل توصيل المودم الخارجي بالكومبيوتر ، ويحتوي على 8 مداخل تستقبل كل منها 0 أو 1 (بايت كامل) ويتم ارسال البيانات تسلسيلاً (بايت ورا التاني) حتى تنتهي البيانات.

قبل كل بايت لا بد من ارسال Start Bit وكذلك بعد الانتهاء ، قد يكون هناك أيضاً ما يعرف باسم Parity Bit في حالة التأكد من الأخطاء وخلافه.

السرعة المتوسطة لهذا المنفذ هو 116 كيلوبت/ثانية ، ولكن مع استخدام ESP أو Super ESP يمكن الوصول إلى سرعات 460 كيلوبت في الثانية.

وبالنسبة ل Parity Bit فباختصار هي طريقة لاكتشاف الأخطاء حيث يمكن عد الأصفار الموجودة في الرسالة ، وفي حالة كونها رقم زوجي يتم ارسال صفر ، وفي حالة الفردي يتم ارسال 1 ، وهكذا يستطيع المستقبل التأكد من دقة البيانات بنسبة معينة.

Parity Error Detection

Odd Parity

Data Bits								Parity Bit
1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	1
1	1	0	0	1	1	0	1	0
0	1	1	1	1	1	1	1	0

Even Parity

Data Bits								Parity Bit
1	1	1	1	1	1	1	1	0
0	0	0	0	0	0	0	0	0
1	1	0	0	1	1	0	1	1
0	1	1	1	1	1	1	1	1

Each time new data is stored in a byte, the RAM, hard disk or serial port's parity generation circuit computes the correct value of the parity bit. In the case of even parity, the value of the parity bit is chosen to ensure the number of 1s in the byte is even. In the case of odd parity, the parity bit is given a value that causes the total number of 1s to be odd.

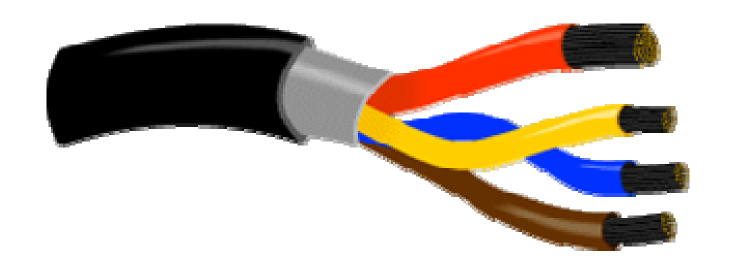
When data is read from a byte, the RAM, hard disk or serial port's parity-checking circuit determines whether the parity bit's value is consistent with the other bits of the byte. If not, one or more bits has changed.

: USB Port

أصبح بفضل سرعته وتوحيد معاييره أحد أشهر المنافذ والتي اتجهت إليها جميع الشركات المصنعة للأجهزة ، حيث نجد سرعة وسهولة التعامل مع هذا المنفذ. فهذا المنفذ يمكنك من ربط حتى 127 جهاز مع الكومبيوتر ، اضف إلى ذلك تمتعه بخاصية hot-swappable والتي تسمح بالربط والتعريف مباشرة مع الكومبيوتر ، اضف إلى ذلك السرعة والتي تبلغ في USB 2.0 حوالي 480 ميجابت / ثانية.



وبخصوص التركيب الداخلي ، فإنه مكون فقط من أربع أسلاك ، واحدة لل VCC وواحدة لل Ground والاثنتان الباقيتان تستخدمان لارسال واستقبال البيانات:



وتنقسم هذه العملية إلى ثلاث أقسام رئيسية:

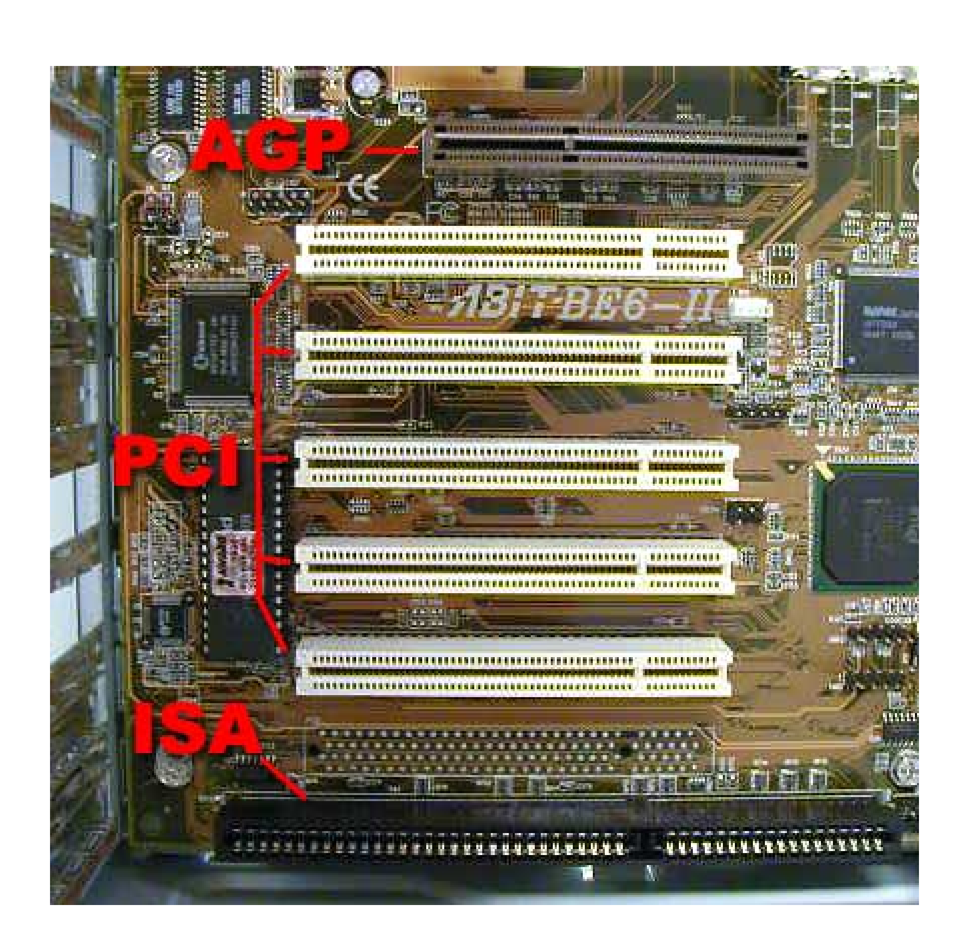
Interrupt : مثل الكيبورد والماوس والتي تقوم بارسال كمية قليلة من البيانات
Bulk : مثل الطابعة حيث يتم ارسال كمية كبيرة من البيانات دفعة واحدة.
Isochronous : مثل الميكروفون والذي يحتاج إلى ارسال كمية متتابعة من البيانات لحظياً ، حيث لا يتم عمل تأكد من الخطأ وهو ما يصب في مصلحة سرعة النقل.

أنوع أخرى من المنافذ:

منفذ PS2 والمستخدم للكيبورد والماوس.



منفذ PCI وهو أعقد أنواع المنافذ واسرعها في نفس الوقت ، وهو على الماذربورد ويتم بناء كارت خاص للتعامل معه ، ولكن يتم التعامل معه ببساطة مثل الأنواع السابقة اجمالاً:



بالطبع يمكن التعامل برمجياً مع جميع هذه الأنواع وهو ما سيتم شرحه في دروس قادمة ، وفي جميع الأنواع يتم الادخال على كل Pin حيث تمثل سلك و bitلادخال البيانات ، أيضاً حتى في PCI ستجد شرائح معدنية تمثل كل منها سلك أو. PIN

في الدرس القادم سوف نتحدث باستفاضة عن Parallel Port والذي سنتابع العمل عليه لاحقاً من خلال دروسنا.

الدرس الرابع : منفذ الطباعة Parallel Port وطريقة التعامل معه بالتفصيل

مقدمة:

كان درسنا السابق مخصصاً للحديث باختصار عن منافذ الكومبيوتر ، هذا الدرس للحديث عن واحد منها ولكن بالتفصيل نظراً لإننا سوف نقوم باستخدامه ضمن هذه السلسلة اضافة لكونه المنفذ الأشهر لعمليات التحكم من خلال الكومبيوتر قبل أن يظهر USB كمنافس قوي لهذا المنفذ ، ولكن ليس للمبتدئين.

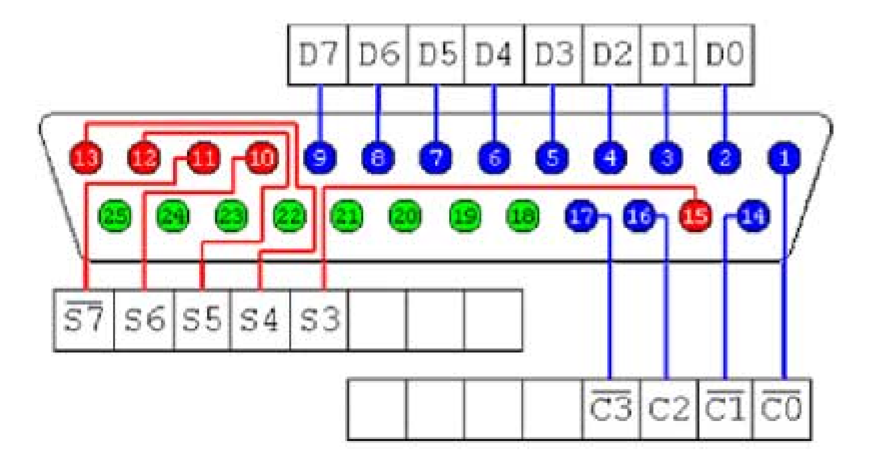
منفذ الطابعة: Parallel Port Interface

يعد منفذ الطابعة أسهل المنافذ للتعامل معها من خلال الكومبيوتر نظراً لكونه أصبح شاغراً بعد ظهور USB واعتماد الأجيال الحديثة من الطابعات عليه ، أيضاً كونه سهل في التعامل معه ولا يحتاج إلى أدوات اضافية وغيره.



وباختصار فإن منفذ الطابعة مكون من خمسة وعشرين Pin أو Hole حسب شكل المنفذ (بمعنى غاطس أو بارز) تقوم بارسال 0 في حالة Close و 1 في حالة - Open حيث أن 1 هي قيمة تقترب من 5 volte في أغلب الأحيان. -

تركيب منفذ الطابعة:



سنبدأ الشرح من خلال الصورة السابقة:
الرقم 1 + الأرقام 14،16،17 تسمى بالـ Control Pins
الأرقام من 2 إلإلى 9 تسمى بالـ Data Pins
الأرقام من 10 إلى 13 + 15 تسمى بال. Statues Pins
الأرقام من 18 إلى 25 تسمى بالـ Ground Pins

Control Pins :
عدد 8 من المنافذ ولكن 4 منها غير ظاهرة تستخدم للادخال والاخراج في نفس الوقت.

Data Pins :
هذه هي ال Pins هي الأشهر والأكثر استخداماً ، وتستخدم دائماً لعرض البيانات بعد توصيلها بجهاز مناسب ، وعددها 8 ، وفي الطابعات كانت تستخدم هذه المنافذ لنقل البيانات إلى الطباعة تمهيداً لعملية طباعتها.

Statues Pins :
تستخدم هذه المنافذ لادخال البيانات ، وعددها الفعلي هو 8 وليس خمسة ولكن هناك ثلاث منها غير ظاهرة ، وفي الطابعات كانت تستخدم لادخال بيانات من الطابعة إلى الكومبيوتر مثل رسائل اشعار انتهاء ورق الطابعة.

Ground Pins :
كأي دائرة كهربائية ، لا بد من وجود قطب موجب وقطب سالب لتعمل الدائرة ، ال Ground هو ما يشكل القطب السالب ، وفي العادة لا يتم استخدام سوى واحد فقط من ال 8 pins لجميع ال Pins الخاصة بالادخال والإخراج.

ارسال البيانات عبر منفذ الطابعة.

لارسال البيانات عبر منفذ الطابعة لا بد من تحديد رقم المنفذ اولاً وعادة ما يكون 387 hومن ثم ارسال قيمة ما بين 0 إلى 255 لتحديد ال Pins التي ستكو عليها القيمة 1 بالشكل التالي مثلاً:
لجعل جميع Pins مفتوحة = 11111111 بالعد الثنائي binary وهو ما يساوي 255.
لجعل Pins الأول والثاني والرابع والسادس مفتوحة = 00101011 بالعد الثنائي binarty وهو ما يساوي 83 = 1+2+16+64 .

*****لمزيد من التفاصيل راجع الدرس الأول الخاص بالتعامل مع نظم الأعداد.**

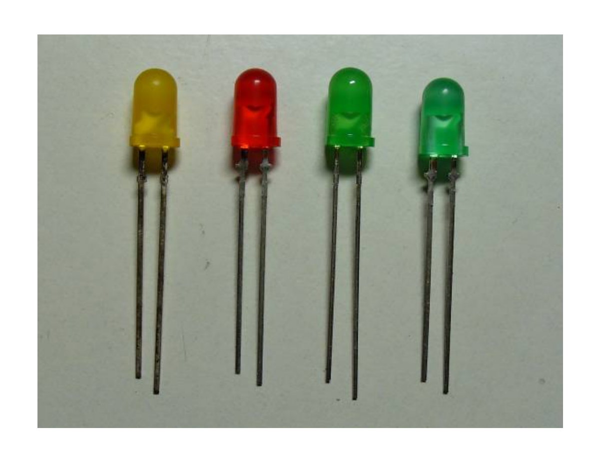
تحديد رقم منفذ الطابعة :

قد يحتوي جهازك على أكثر من منفذ للطابعة اضافة إلى احتمالية كون رقم المنفذ غير الافتراضي ، لذلك من لوحة التحكم - Control Panel النظام System -الأجهزة - Hardware ادارة الأجهزة - Device Manager المنافذ Ports (COM & LPT) - منفذ الطابعة - Printer Port (LPT1) الخصائص Properties ، ثم من خلال المصادر Resources ثم Resources Setting سوف تجد الرقم المخصص للمنفذ الخاص بك.

في الدرس القادم سوف نتعلم كيفية برمجة المنفذ ، ولكن حتى الدرس القادم تستطيع عمل هذه التجربة البسيطة:

تجربة:

قم بشراء واحد من ال LED's من أي محل للالكترونيات ، وباختصار فال Leds هي عبارة عن انوار كهربائية بسيطة تنير عندما يمر فيها فرق جهد = 5 ، وفي خلال دروسنا القادمة سوف نتعرف تفصيلاً على تركيبه هو وبعض الأجزاء الأخرى التي سوف نقوم باستخدامها.



سعر ال LED الواحد لا يتعدى 20 قرشاً مصرياً (أقل من ربع جنيه (ويمكنك شرائها من أي محل للالكترونيات.

خطوتنا الثانية هي معرفة أن لل LED توصيلة للسالب وأخرى للموجب ، قم بتوصيل السالب بواحد من ال Grounds في منفذ الطابعة لديك وقم بتوصيل الموجب في أحد ال Data Pins وليكن الأول لديك ، ومن ثم قم بعمل برنامج سريع يقوم بفتح واغلاق هذا ال Pin .

إذا كنت مبرمج net. فقم أولاً بعمل import أو - using حسب VB أو - #C ل : System.Runtime.InteropServices ومن ثم قم باستيراد المكتبة impout32.dllبالشكل التالي - سي شارب: -

```
[DllImport("inpout32.dll", EntryPoint="Out32")]
```

ومن ثم قم بتعريف ال function التالية - مبرمج سي شارب. -

```
public static extern void Output(int adress, int value);
```

أو مبرمج: VB.net

```
public static extern sub Output(int adress, int value)
end sub
```

ومن ثم قم بكتابة الرقم الذي يحدد ال Pins التي تريدها مفتوحة وليكن مثلاً الأول فقط وقم بارساله باستخدام Output بالشكل التالي:

```
Output(888, 1);
```

مبرمج VB.net ؟؟؟؟ فقط قم بازالة ال ; من آخر السطر.

وحتى إذا كنت مبرمج Visual Basic 6 فهذا الأمر لك أيضاً ولكن وفر وقتك وجهدك وتحول إلى - net. دعاية مجانية 🙂-.

قم بعمل Module وضع هذا الكود داخله:

```
Public Declare Function Inp Lib "inpout32.dll" _
Alias "Inp32" (ByVal PortAddress As Integer) As Integer
Public Declare Sub Out Lib "inpout32.dll" _
Alias "Out32" (ByVal PortAddress As Integer, ByVal Value As Integer)
```

ومن ثم في زر الأمر (فتح) .

```
Out "&H378", 1
```

زر الأمر (اغلاق).

```
Out "&H378", 0
```

حيث يتم فتح واغلاق كامل ال Pins في كل خطوة.

تنبيه : في حالة كونك تعمل على Windows XP فسوف تحتاج لتنزيل مكتبة جديدة في الغالب بدلاً من الافتراضية inpout32.dll وهي موجودة في رابط الدرس هنا :

http://vb4arab.com/vb/showthread.php?t=1531

الدرس الخامس : برمجة منافذ الكومبيوتر

كنا في الدرس السابق قد تعلمنا كيفية كتابة الكود ، في هذا الدرس سنتعلم كيف يتم تنفيذ هذه الأوامر فعلياً قبل أن ننطلق في الدرس القادم في عالم الهاردوير.

يتم التعامل مع المنافذ في الكومبيوتر أو مع الهاردوير عموماً مروراً بأربعة مراحل:

1- Device Driver : وهو عبارة عن برنامج خاص بالتعامل مع هاردوير معين يتم انتاجه من قبل الشركة المصنعة لهذا الجهاز ، حيث أن الشركة المصنعة للطابعة تقوم مثلاً بعمل البرنامج الخاص بالطابعة بحيث يحتوي على مجموعة من الأوامر مثل Print Page : ، Print Pages ... الخ ، حيث يقوم المبرمج أو نظام التشغيل بالتعامل مع هذه الأوامر فقط دون الدخول في تفاصيل كيفية عمل هذه الأوامر والتي تهتم بها فقط الشركة المصنعة.

2- Operating System : في اصدارات قديمة من نظام التشغيل Windows كان يمكن للمبرمج التعامل مباشرة مع منافذ الكومبيوتر أو تحديداً مع ال Device Driver مباشرة ، وكان هذا في DOS و Window 9x حيث كان الموضوع يقتر على مجموعة من ال API's للتعامل مباشرة ، ولكن بعد هجوم CIH عام 98 الشهير بدأت نظم التشغيل تحمي الهاردوير عن طريق عدم السماح لأي شخص بالتعامل معه مباشرة.
مع الاصدارت التالية تم فقط تعريف نظامين للتعامل مع الهاردوير ، 0 و 3 ، حيث يقوم المبرمج بالعمل في 3 في حين يعمل الهاردوير وال Kernel في ال 0 أو ما يعرف باسم الحلقة 0 - ring 0 - حيث لا يمكن للمستخدم الوصول مباشرة إلى الهاردوير إلا عن طريق kernel وهو المرحلة الثالثة من عملنا.

3- Kernel Mode Driver : وهو برنامج مخصص يمكنه نقل البيانات من وإلى ال Device Driver، ويتم استخدامه في نظم التشغيل التي لا تسمح بالوصول المباشر إلى الهاردوير.

أشهر أنواع ال KMD كمبرمج ويندوز هي - inpout32.dll ارفقناها في الدرس السابق- والتي تحتوي على جزء منها Kernel Mode Driver والجزء الآخر خاص بدول API العادية التي تحتويها المكتبة.

5- Programming : المرحلة الأخيرة من هذه العملية ، حيث يتم من خلالها عمل البرنامج الخاص بالتعامل مع المستخدم من جهة ومع الهاردوير من جهة أخرى ، لن تفرق لغات البرمجة كثيراً في هذه النقطة ، في الدرس السابق شرحنا كيفية التعامل من خلال C#,VB.net وفيجوال بيسك 6 ، هنا نضيف انه في ال Java سوف نستخدم Javax.comm.parallelport ، في ال PHP سوف نستخدم إحدى السيرفس والتي تسمى lptout ،بخصوص ++C فهي مثلها مثل #C تحت الويندوز ، أما ++C في نظام التشغيل لينوكس فسوف نستخدم *ioperm*الموجودة في. <asm/io.h>

الدرس السادس : مدخل إلى الهاردوير - الجزء الأول

بعد ان تعرفنا في الدروس السابقة على كيفية اخراج البيانات عبر منفذ الطابعة بالكود ، واصبح بمقدورنا التحكم في ال PINS 8 المخصصين لل Output ، جاء الدور الآن على كيفية قراءة هذه القيم والاستفادة بها في العمليات المختلفة في الجهة المقابلة (جهة الهاردوير.)

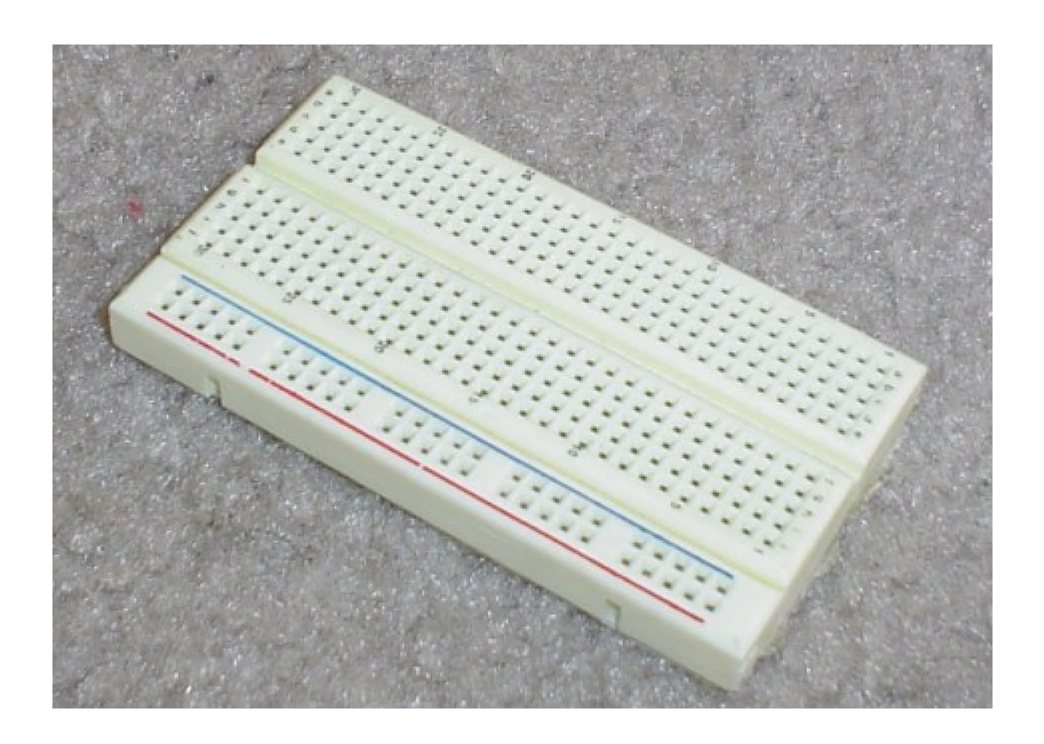
أبسط طرق قياس هذه القيم هي استخدام LED واحدة ، يتم توصيل طرفها الموجب بأحد ال Pins الموجودة لل Output وطرفها السالب بأحد ال Grounds الموجودة ، وستضيء عندما يكون هذا ال Pin مفتوحاً.

طريقة أخرى سهلة للقراءة وهي استخدام فولتاميتر Volt Meter حيث يتم توصيله بال Pin المطلوبة وال Ground أيضاً وقراءة فرق الجهد الناتج:



كانت هذه هي الطرق التي تعلمناها في الدرس الماضي ، والهدف منها فقط قراءة والتأكد من عمل البرنامج ، أما في هذا الدرس ولمدة ثلاث دروس قادمة فسوف نتعرف على بعض الهاردويرس المستخدمة لبناء الدوائر المختلفة ، في هذا الدرس سوف نتعرف على أساساتها...

أولاً : اللوحة الأم:breadboard



تستخدم هذه اللوحة لتوصيل وتثبيت الدوائر الكهربائية ، وهي الحل الأبسط والامثل للمبتدئين ، حيث تجد أن كل خط عرضي - 5 خانات في الصورة مثلاً والمرقمة بحروف - تحمل قيمة واحدة بمعنى أنك لو ادخلت قيمة في أي منهم وقرأتها في الآخرين فسوف تظل هي نفس القيمة.

نستفيد من ذلك فيما لو رغبنا في قراءة قيمة مثل Ground لأكثر من دائرة فإننا سنضع ال Ground في خانة ونقرأ من اربع خانات لأربع دوائر مختلفة ، وهكذا.

وبخصوص الأشرطة السفلية والعلوية في الصورة دعك منها مبدئياً ، فهي نفس ما سبق ولكن خطوطها بالطول وليس بالعرض للبيانات التي تحتاج لقرائتها عدد كبير من المرات.

سعر هذه اللوحة حوالي 10 جنيهات أو أكثر قليلاً ، لكن هناك حل آخر وهو ما يسمى بالفيبر وهو أرخص ولكنه أفضل في التعامل وأصعب نسبياً ، وفي كلا الحالتين يتم نقل البيانات باستخدام أسلاك عادية ، الصورة التالية كمثال:



أما مثلما يحصل في ال MotherBoards وهي ما تسمى بالدوائر المطبوعة Printed Circuits حيث يتم نقل البيانات باستخدام سيلكون مطبوع ويتم تثبيت الدوائر فهي الطريقة الأمثل للمحترفين ولكن طباعتها إلى حد ما مكلفة ، كما أنه لا يمكنك تغييرها لاحقاً.

ثانياً : مصدر الكهرباء Adapter - مثلاً: -

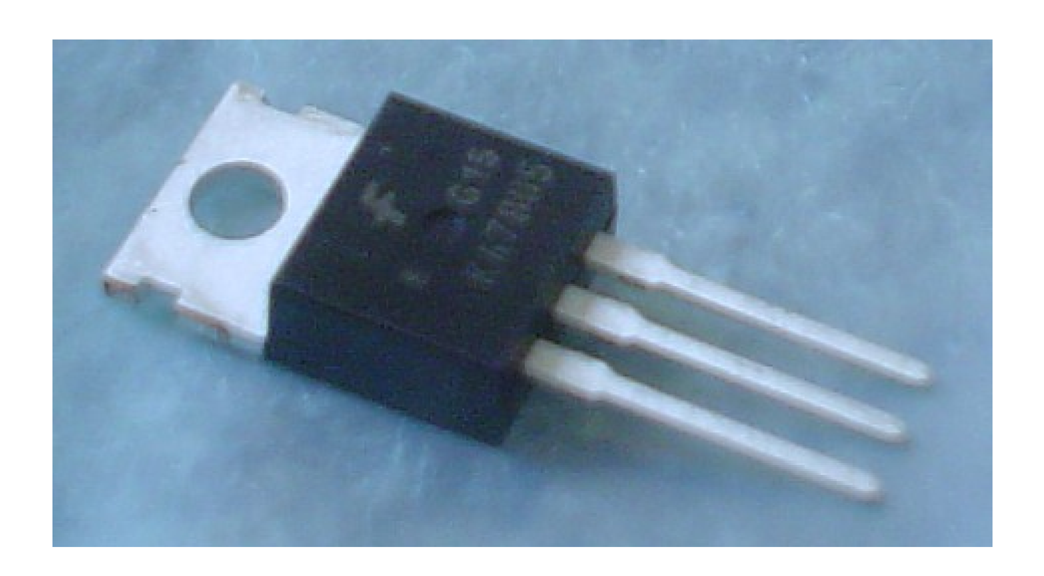
في حالة تعاملك مع أي دائرة كهربائية فإنك تحتاج إلى VCC اضافة إلى Ground وحينما كنا نتعامل مع ال LED كنا نقرأ هذه البيانات عن طريق الكومبيوتر ، ولكنك في دوائر أخرى ستحتاج إلى مصدر خارجي للتيار الكهربائي ، يمكن أن يكون بطارية 9 فولت ، 20 فولت ، أو adapter يتم توصيله بالكهرباء بواسطة Adapter مثل الذي في هذه الصورة:



وبالطبع فإنك ستشتري واحدة يمكنه تزويدك بفرق جهد عادة ما يتراوح بين 1.5 و 12 فولت يمكنك التحكم فيها كما تريد ، كما يمنحك قطبين سالب وموجب.

سعر هذا المحول حوالي 10 جنيهات مصرية.

ثالثاً : منظم الكهرباء: Regulator



الهدف من هذا المنظم هو الحفاظ على فرق جهد ثابت بمعنى انك لو وضعت 5 v regulatorفسوف يحافظ لك على فرق الجهد 5 فولت حتى لو كنت تستخدم بطارية 9 فولت مثلاً ، حيث اننا دائماً نحتاج إلى قيمة 5 فولت في أغلب الاحيان .

طريقة استخدامه ، يتم وضع ال Ground القادم من مصدر التيار سواء الكومبيوتر أو المحول أو البطارية في المنتصف وتتم قرائته من المنتصف ايضاً ، أما ال VCC فيتم وضعه في اليمين ثم قرائته من اليسار وادخاله على أي جهاز مطلوب.
سعر المنظم حوالي 3 جنيهات.

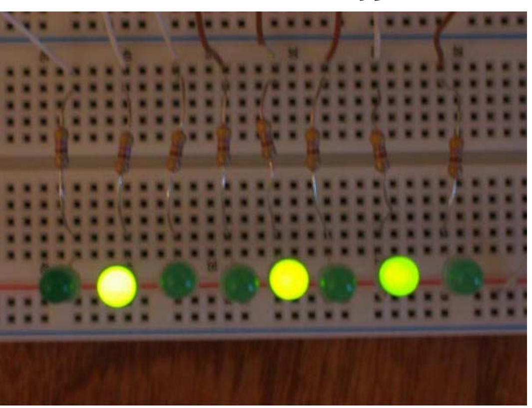
المقاومات Resistors والترانزستورات: Transistors

تم شرحهما سابقاً ، راجع الدرس الأول لمزيد من التفاصيل...

مثال سريع:

سنحاول الآن تطبيق مثال سريع ومرة أخرى من خلال LED's ، مقاومات ، وسنقوم بالتطبيق مرة من خلال الكومبيوتر ومرة من خلال مصدر كهرباء خارجي سواء أكان بطارية أم محول.

سنستخدم لوحة نقوم بتثبيت LED عليها ، وقبلها سنقوم بوضع مقاومة صغيرة لكي تحافظ لنا على ال LED ، يتم وضع المقاومة قبل ال VCC أما ال Ground فيظل كما هو مثل هذه الصورة:



طبعاً لن تتم القراءة مباشرة في حالتنا من الكومبيوتر أو من مصدر التيار ، بل سوف نربطها أولاً باستخدام Regulator وكما تعلمنا من كيفية قرائته من اجل الحفاظ على فرق الجهد ثابتاً ولأجل التعود عليها حيث سنحتاجها في دروس قادمة.

هذا التطبيق مشروح تفصيلاً على هذا الرابط ، يمكنك الاطلاع عليه للاستفادة وللشرح الموسع:
http://www.codeproject.com/csharp/csppleds.asp

في الدرس القادم سوف نبدأ بالتوسع في الهاردويرس المستخدمة بكثرة في عمليات الاخراج Outputs قبل أن نعود للهاردوير المستخدم للادخال Input ومن ثم ننتقل إلى الدوائر الالكترونية المستخدمة في عملية المعالجة قبل أن ننطلق إلى شرح لبعض أكثر الدوائر شيوعاً..

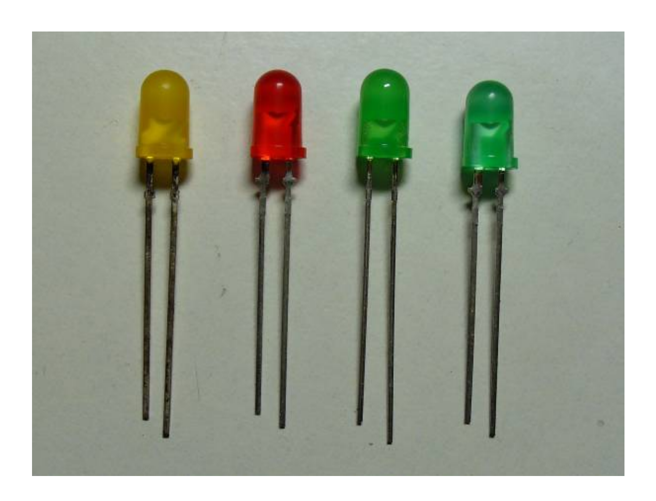
ولا داعي لإن اذكر بأنني لن استوفي كل شيء ، فقط سنطرح النقاط الأساسية والتي يستحسن ان تكون مطلعاً عليها كونك مبرمج كومبيوتر ، أما مهندسي الكومبيوتر فلهم مجالات متعددة ومواضيع كثيرة لن اتطرق لها ضمن الدروس..

الدرس السابع : مدخل إلى الهاردوير - الجزء الثاني

شرحنا في الدرس السابق اساسيات العمل من خلال الهاردوير ، في هذا الدرس سوف نقوم بعرض بعض الوسائل التي تستخدم في عمليات عرض البيانات.

1- LED’s.

سبق شرحها أكثر من مرة وباختصار شديد فهي عبارة عن (لمبة) صغيرة تفتح في حالة كون ال. VCC=5

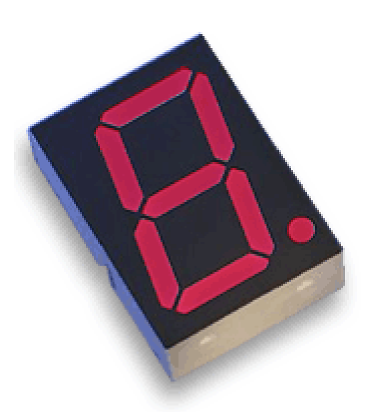


2- Buzzer :

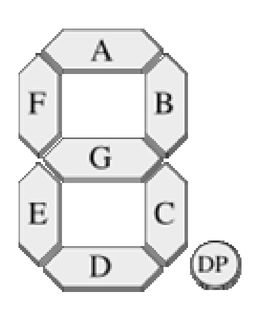
في ال LED's كنا نستخدم الاضاءة كدليل على مرور التيار الكهربائي ، أما في Buzzerفهو يصدر صوتاً وقت أن يكون ال: VCC=5



3- Seven Segment :



مثل الصورة السابقة ، فهو يقوم بعرض رقم اعتماداً على اضاءة واغلاق عدد من ال inputs الخاصة به ، وهو لا يستقبل مدخل واحد بل يستقبل عدد 8 مدخلات كل منها يمثل اضاءة من ال 8 الموجودة + النقطة حيث يمكن تشكيل الرقم المطلوب ، بالشكل التالي مثلاً:

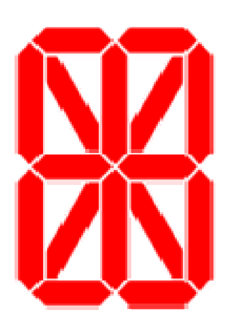


حيث يمكن احتساب اضاءة كل واحد منهم على حدة وحسب الرقم المدخل - بعد تحويل الرقم إلى Binary طبعاً - بمعادلات مثل هذه المعادلات:

A = D'C'B'A' + D'C'BA' + D'C'BA + D'CB'A + D'CBA' + D'CBA + DC'B'A' + DC'B'A

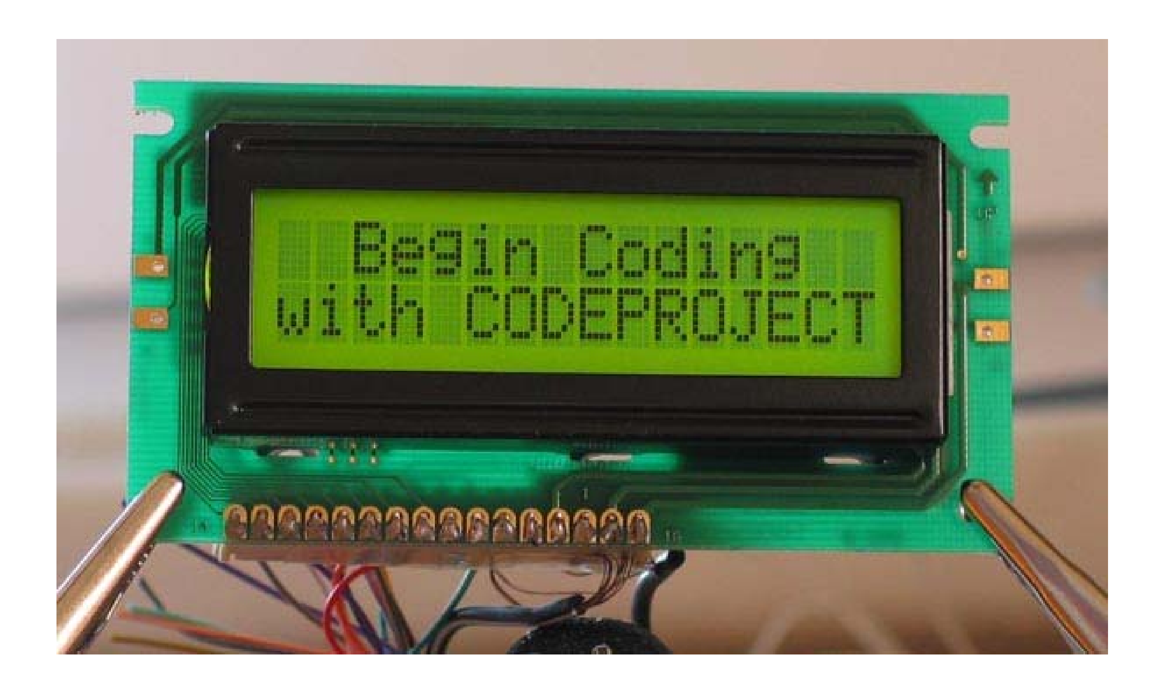
وهكذا لكل خانة منهم A,B....,DP معادلة خاصة بها.

هناك بالطبع اصدارات تستخدم خانتين ، ايضاً هناك ال Fourteen Segment وال Sixteen Segmentوهي مثل تلك التي في الصورة حيث توفر خيارات اضافية لكتابة الخانات:



4- ال Screens :

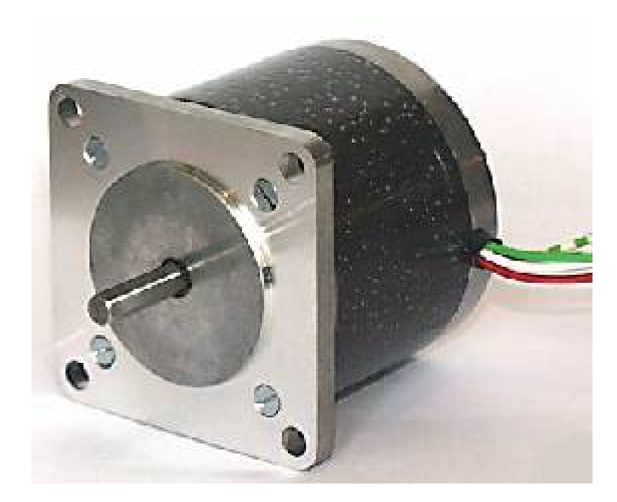
أكثر تعقيداً بالطبع من ال Seven Segment وعائلتها حيث تسمح لكل بكتابة عدد معين من الخانات يزيد وينقص حسب نوعية الشاشة.



5- ال Stepper Motors وباقي المحركات :

تعتمد على الحركة للاستفادة من التحكم ، وهي مجموعة من المحركات التي تستقبل بيانات رقمية تحدد سرعة وحركة الموتور ، بالطبع هناك الكثير من

التفاصيل في هذا الجزء وكيفية التحكم به والعمل من خلاله وهكذا...

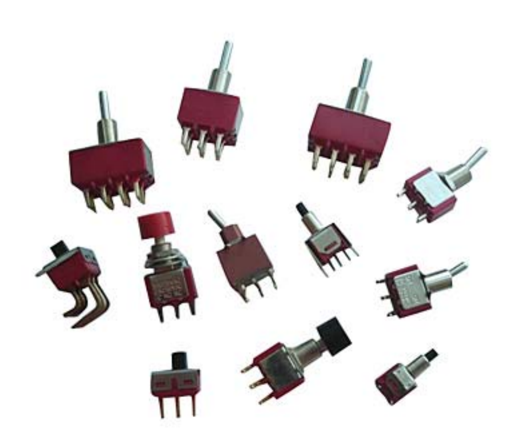


نكتفي بهذا القدر اليسير في هذا الدرس ، في الدرس القادم سوف نواصل ولكن مع عدد بسيط من وسائل الادخال.

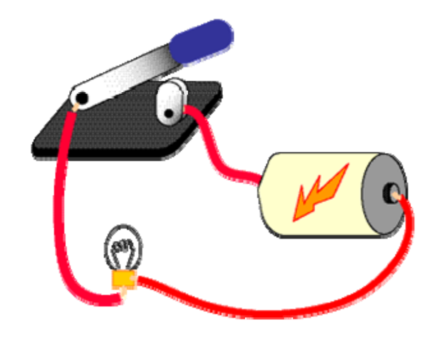
الدرس الثامن : مدخل إلى الهاردوير - الجزء الثالث

نواصل مبادئنا السريعة التي بدأنا فيها قبل درسين من الآن ونحاول فيها تقديم نبذة مختصرة قبل الشروع في التطبيق العملي ، في هذا الدرس سوف نعرض سريعاً عدداً من وسائل إدخال البيانات.

أولاً: Switches :

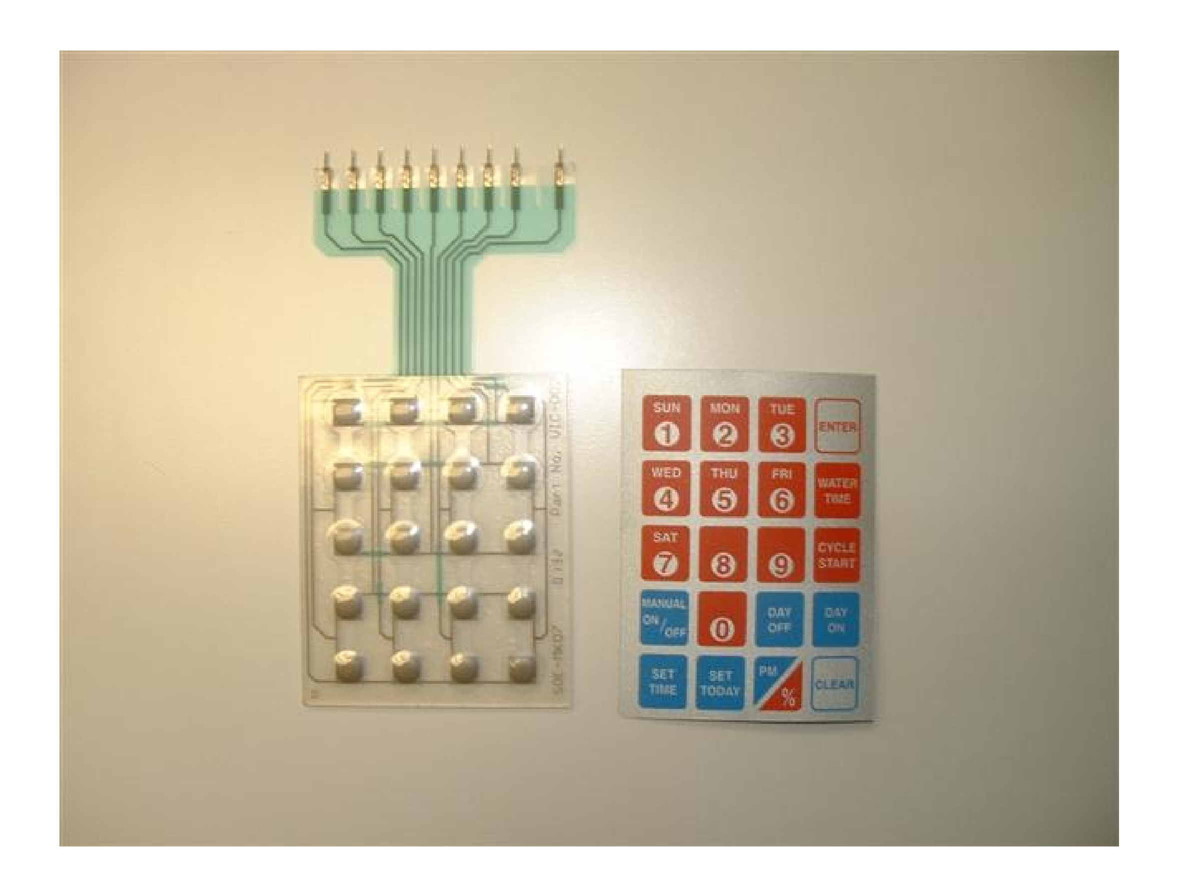


توضح الصورة السابقة عدداً من ال Switches لامختلفة ، وتشترك كلها في مبدأ قطع التيار ، حيث أنه في حالة ما فإن التيار يمر بين طرفي السويتش في حين أنه في الحالة الأخرى فإنك السلك يعتبر مقطوعاً ، تستخدم بطريقة توصيل تيار كهربائي مستمر ووقت أن نحتاج إلى ادخال بيانات إلى الكومبيوتر فإننا نقوم بفتح السويتش المتصل مع ال Parallel Port مثلاً ، مثل استخدامك لأي مفتاح للكهرباء أو للمياه ، تماماً مثل الصورة التالية:



هناك عدد من السويتش (ثنائي - رباعي) وأكثر تعتمد على نفس المبدأ ولكن تقوم بفتح أكثر من خط في نفس المرة ، أو في نفس الأداة تجد التحكم بأكثر من سلك.

2- لوحة المفاتيح KeyPad :



على اختلاف انواعها فهي تشترك في مبدأ واحد ، فمثلاً زر رقم 7 يتم تمثيله على keypad مرتبطة ب 8 خانات بالشكل التالي : 00000111 ، ويمكن في هذه الحالة السماح لبعض الحروف حيث يتم تمثيل A على سبيل المثال 11000100 - مثال فقط. -

ثالثاً : المجسات: Sensors

هو عالم متوسع متشعب مليء بالابتكارات والاختراعات والتقنيات المتجددة دائماً ، حيث تشكل المجسات الحواس التي يمكن للجهاز أي يستشعر بها العالم الخارجي ، فمثلاً يمكن استخدام مجسات لقياس درجة الحرارة ، سرعة الدوران ، وجود ماء من عدمه ، انذار الحريق ، تتبع الاضاءة ، الحرارة ، الالوان ... الخ ، وتدخل المجسات في كثير من الصناعات مثل الأمن الصناعي ، عالم الروبوتات والتحكم الآلي والقيادة وخلافه.

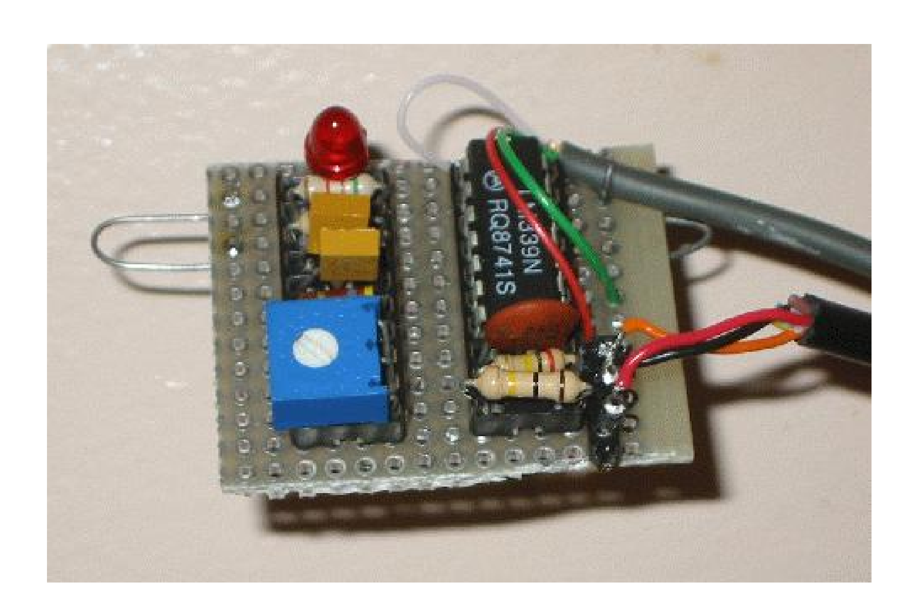
طبعاً لا يتم بداية تمثيل البيانات بصورة رقمية حيث يتم في العادة قراءة قيم المجسات على شكل مقاومة مثلاً أو ما شابه ، ولكن يتم تحويلها لاحقاً إلى صورة رقمية يستطيع الكومبيوتر قراءتها.

أمثلة : سينسور ضوئي:



يقوم بقراءة التغير في شدة الإضاءة.

أمثلة : سينسور مائي:



يصدر اشارة في حالة وجود مياه تلامس الجزء الحساس منه.

أمثلة : سينسور الحريق:



يصدر اشارة في حالة وجود لهب.

وغيرها من الأمثلة.

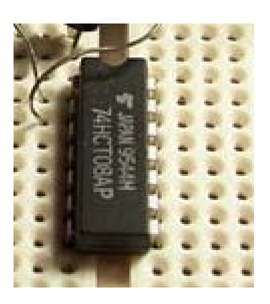
في الدرس القادم سوف ننتقل إلى دوائر المعالجة قبل أن نبدأ في التطبيقات العملية.

الدرس التاسع - وسائل المعالجة

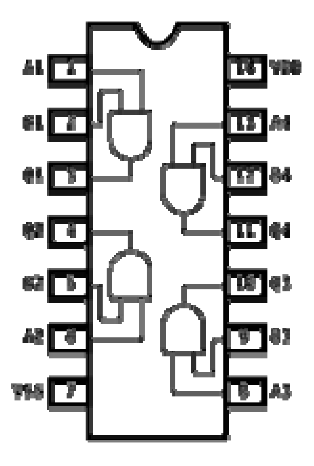
بعد أن تعرفنا على طرق الادخال والإخراج ، سنعود هنا في درسنا الأخير ضمن سلسلة الدروس النظرية للتحدث عن طرق معالجة البيانات ، قبل أن نواصل في سلسلة دروسنا القادمة مع مثال حي وتطبيقي خطوة بخطوة.

أبسط وسائل المعالجة : المعالجات المنطقية: Logic Gates

مثل And و Or و Not وخلافها ، وباختصار يمكن استخدامها في عمليات الجمع والطرح ، وعكس الاشارة وخلافه خارج نطاق الكومبيوتر ، ويتم عادة تجميعها ضمن Gates تستقبل مصدر كهربائي VCC و Ground اضافة إلى عدد من المدخلات وبالتالي عدد من المخرجات أيضاً ، على سبيل المثال دائرة And تكون بهذا الشكل مثلاً:



أما تركيبها الداخلي فيكون بالشكل التالي مثلاً:



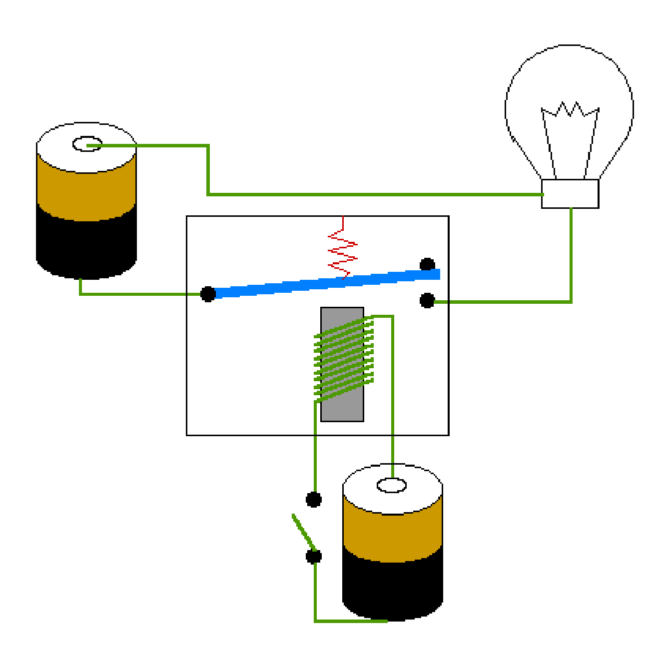
حيث أن A1 مع B1 يكون ال Output الخاص بهم على Q1 مثلاً...

هناك وسائل أخرى لاتخاذ القرار مثل Buffer على سبيل المثال ، وهو يسمح للتيار (أياً كانت قيمته) بالمرور من نقطة1 إلى نقطة2 في حالة كون المدخل 1 على سبيل المثال ، ونستفيد منه بعمل متحكمات.

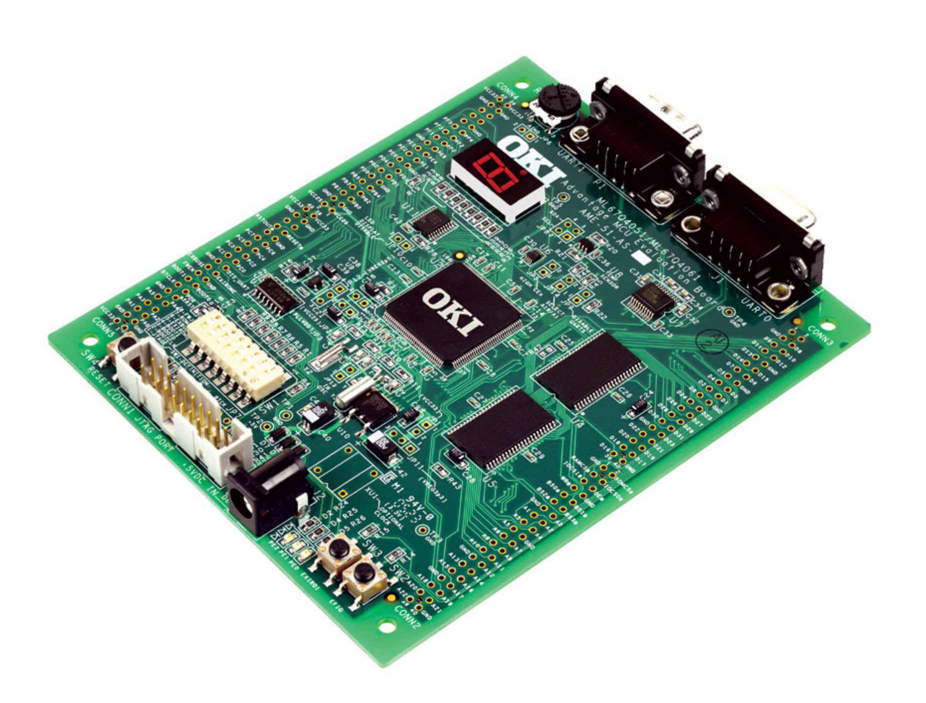
المتحكم الأكثر تعقيداً هو ال Relay حيث نلحظ في ال Buffer أنه لا يمكن له التعامل سوى مع فرق جهد لا يتعدى 12 ، أما ال Relay فهو قابل للتحكم بفرق جهد يزيد حتى عن 220 فولت ، ولذا يعد أحد أحجار الزاوية في أي تطبيق صناعي حيث يستخدم مثلاً للتحكم في تشغيل واطفاء الأجهزة الكهربائية بدلالة فرق الجهد القادم من الكومبيوتر ، وسنقوم باستخدامه إن شاء الله في مرحلة التطبيق العملي ، هذه صورة ال Relay على أية حال:



وهذه هي فكرة عمله:



هناك الكثير من وسائل التحكم في العمليات ، ولكن سننتقل مباشرة إلى أكبرها وهو MicroController ، وهو ليس اكبرها اعتباطاً ولكن لإنك تقوم ببرمجته لينفذ لك أي نوع من المهام التي ترغب فيها ، وعادة ما يكون لل MicroControllerوسائل ادخال واخراج ويحتوي على ذاكرة ومعالج مختلف الكفاءة والقدرة حسب نوع ال MicroController وسعره ، لن نستخدمه في أمثلتنا القادمة ولكن ربما بعد انتهائنا من سلسلة الدروس هذه قد تكون هناك دروس خاصة ببرمجة ال MicroController ، هذه صورة ل MicroController على سبيل المثال:



الدرس العاشر : تطبيق تشغيل الأجهزة الكهربائية من خلال الكومبيوتر

مقدمة:

هذا هو درسنا التطبيقي الأول ، في هذا الدرس سنتعلم كيفية التحكم بالأجهزة الكهربائية من خلال الكومبيوتر .

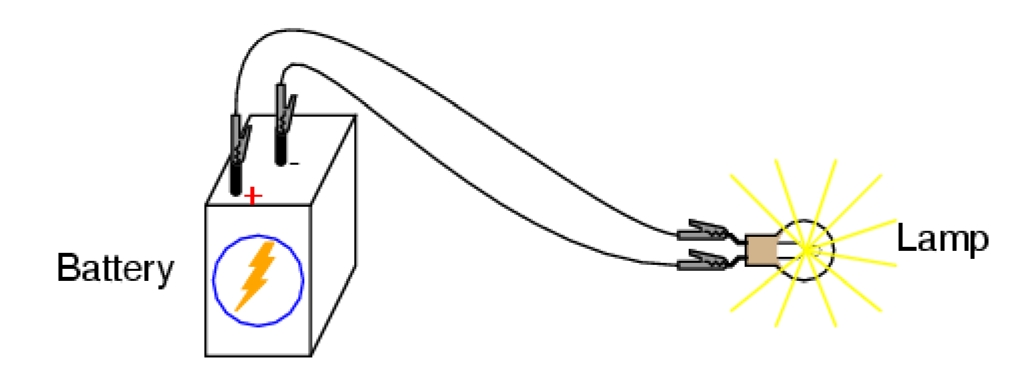
والآن كل ما عليك هو تجهيز أحد أجهزة المنزل الكهربائية (جرس - مصباح - مروحة - أو أي جهاز كهربائي مكيف - ثلاجة ولكن أوعى تبوظها 🙂) .

القاعدة العامة في جميع الأجهزة السابقة على اختلاف انواعها أنها تستخدم كهرباء سواء كانت من الكهرباء العمومية 220 فولت ، أو حتى أجهزة تستمد كهرباء من مصدر أقل مثل بطارية ، شخصياً قمت باحضار مصباح عادي وسأطبق عليه الدرس ، يمكنك عمل نفس الخطوات مع أي جهاز كهربائي كما أسلفت ولن تجد فرقاً واحداً.

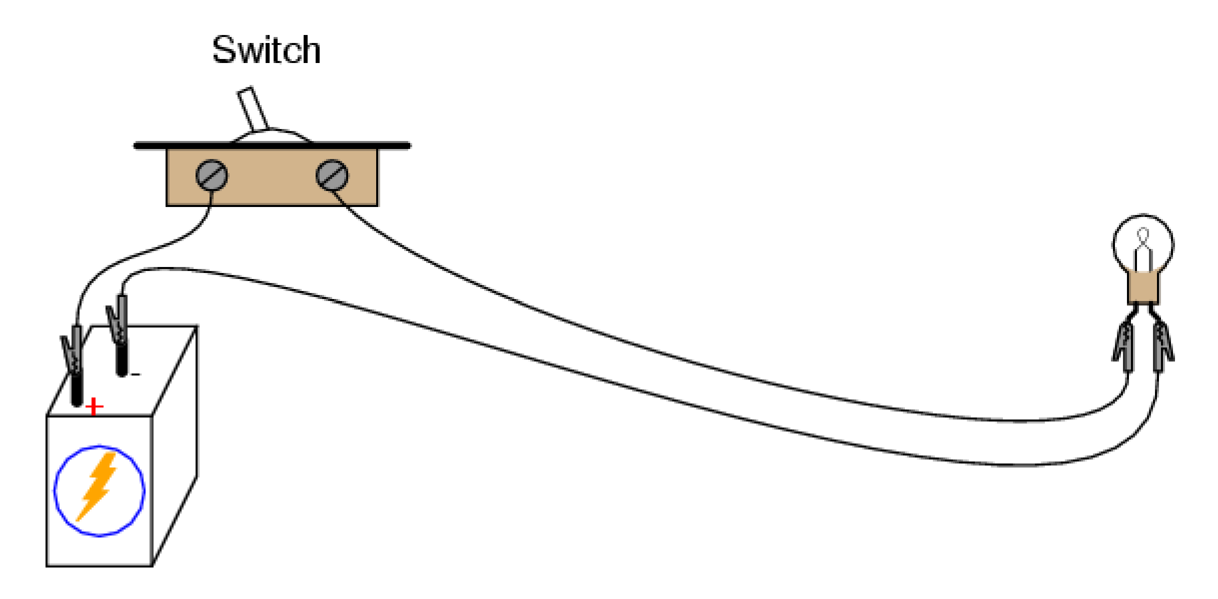
لا تنس أن تتوخى الحذر خلال تعاملك مع الكهرباء حيث سنتعامل في هذا الدرس ولأول مرة مع كهرباء عالية الجهد 220 أو 110 فولت ، وهو خطر جداً لإننا سنتعامل مع أسلاك عارية ، لذا كن على حذر شديد ، حاول دوماً ان تكون في مكان غير مبتل وعلى كرسي خشبي أو مكتب خشبي مثلا بعيداً عن الأرض ، الحرص واجب برغم أن الموضوع سهل تماماً ويتعامل أغلبنا مع الكهرباء بصورة دورية ونتعلم درس الدائرة الكهربائية ابتداء من الصف الرابع الإبتدائي ، ولكن لا أريد أن أكون سبباً في مضرة أحد ، أيضاً خلال مراحل التطبيق احرص دوماً على كون الدائرة مفصولة عن التيار الكهربائي ، وبعد انتهائك من العمل واستعدادك لتجربة شيء ما قم بتوصيل الكهرباء بحذر شديد ، حاول ألا يكون هناك أحد آخر في الغرفة (طفل صغير) لكي لا يعبث بالكهرباء. بعد انتهائك لا تنس فصل التيار الكهربائي ووضع الدائرة في مكان آمن.

البداية (الكهرباء) .

كما أوضحت في المقدمة ، قمت باحضار مصباح كهربائي عادي جداً وقمت بتركيب وصلة كهربائية في طرفه لأتمكن من وصلها بكهرباء المنزل ، من البديهي كما ترى أن تعلم أن السلك الموصل للتيار هو عبارة عن دائرة (سالبة وموجبة) مع التيار الكهربائي المصدر بالشكل التالي - تخطيط الصورة على مصدر تيار كهربائي (بطارية) :



لا يخفى عليكم أيضاً أننا نستطيع اضافة زر تحكم (ON/Off) في المنتصف بحيث يصبح تخطيط الدائرة بالشكل التالي:



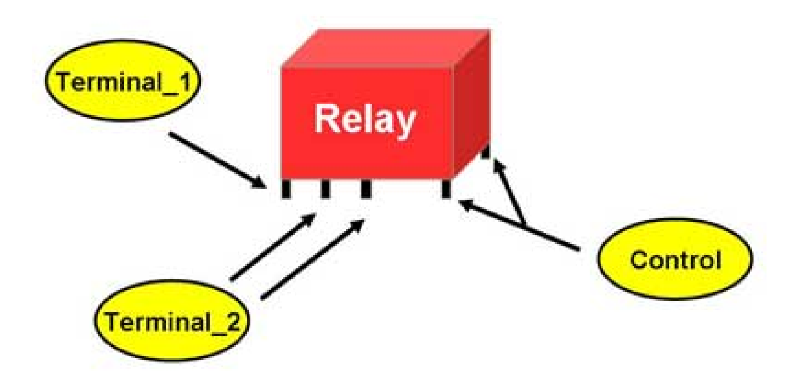
وبهذا نكون قد كونا دائرة كهربائية كاملة ، فيها مصدر تيار كهربائي ، زر التحكم ON/OFFتتصل في النهاية بمصباح يتم فتحه واغلاقه عن طريق التحكم بهذا الزر ، وحتى هذه اللحظة ، نكون بعيدين تماماً عن الكومبيوتر فكل الأدوات التي تحتاج لها هنا تستطيع توفيرها من أقرب محل للأدوات الكهربائية.

الخطوة الثانية ، مع ال: Relay

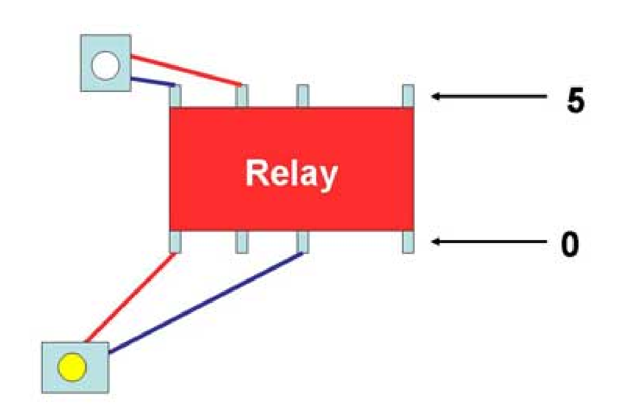
كما ترى فكل ما نحتاج إليه في نهاية هذا الدرس أن نستطيع التحكم بالدائرة الكهربائية السابقة ولكن بدلاً من استخدام ايدينا لفتح واغلاق الدائرة نود استخدام الكومبيوتر لهذا الغرض ، هنا يأتي دور المتحكم. Relay

ببساطة شديدة ، يعتبر ال Relay عبارة عن عنصر تحكم يسمح (بمرور - عدم مرور التيار الكهربائي) مثله مثل المفتاح السابق ، ولكن الفارق الوحيد أن المفتاح السابق تتغير حالته طبقاً لتغيير المفتاح يدويا ، أما ال Relay فتتغير

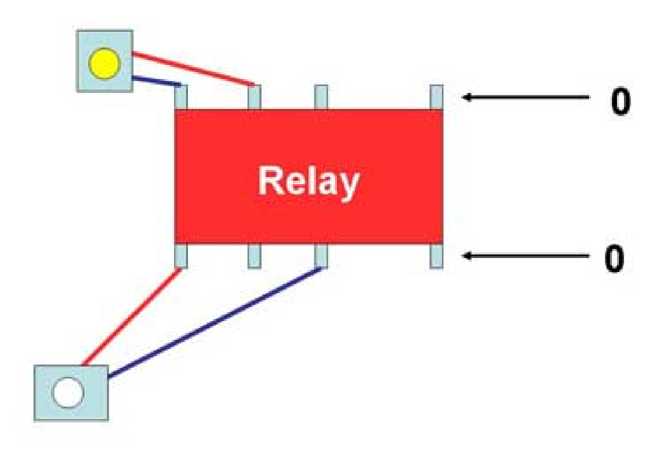
حالته بناء على ال Input الذي يدخل إليه (صفر أو واحد) وهو ما يناسبنا تماماً في عالم الكومبيوتر ، المخطط التالي لشكل ال: Relay



في الصورة السابقة ترى أن هناك pins مخصصة لاستقبال التحكم Control ، يمكن عمل ذلك بأحد الأشكال التالية:



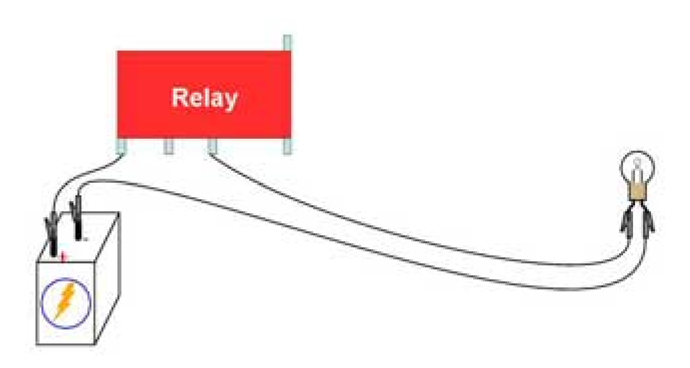
أو



حيث يتم وصل الطرف الأول بجهد أرضي والطرف الثاني بجهد 5 Vليعمل ال Relayويقوم بنقل التيار الكهربائي بين الطرف الأول والثالث ، في حالة كونه 0 Vيتم نقل التيار الكهربائي بين الطرف الأول والثاني.

التوصيل مع الدائرة الكهربائية:

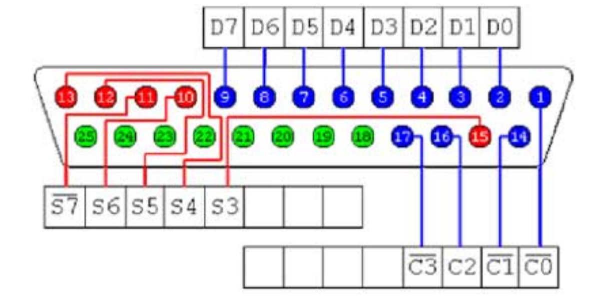
كل ما سنحتاج إليه هو توصيل اطراف الدائرة والتي كانت في مكان المتحكم اليدوي Switch بالطرف الأول والثالث لل Relay ، لتصبح بالشكل التالي:



وهكذا تكون دائرتنا الكهربائية قد اكتملت ، في انتظار حصولها على أوامر التحكم من خلال الكومبيوتر.

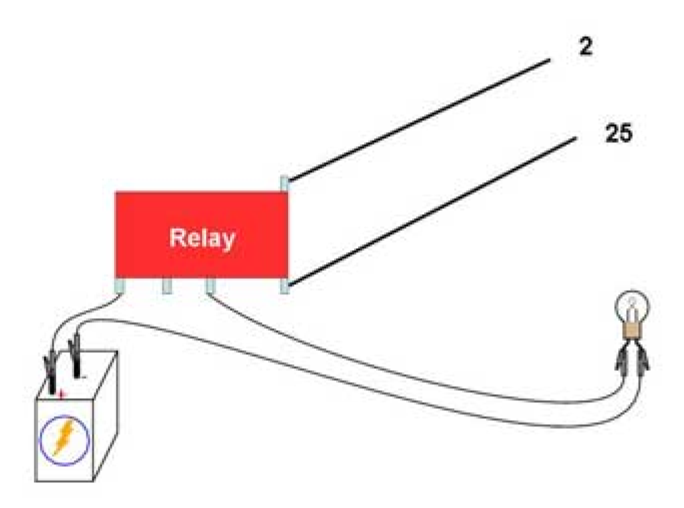
أوامر التحكم من خلال الكومبيوتر:

باختصار شديد ، سنقوم بارسال الإشارات الكهربائية عبر منفذ الطابعة Parallel Port، والذي له التركيب التالي كما أوضحنا سابقاً :



وكما أسلفنا ، فكل ما يهمنا هنا في هذه الدائرة الكهربائية هي ال pins باللون

الأزرق والتي تمثل ال VCC والتي باللون الاخضر والتي تمثل الجهد الأرضي Ground، سنقوم بأخذ سلك من pin رقم 2 والمسماه D0 لتكون المصدر الأول للكهرباء لدينا ويتم ادخالها على VCC الخاص بال Relay ، فيما نحصل على الجهد من اي من ال pins وليكن من رقم 25 ونقوم بادخالهما سوية على ال Relayبالشكل التالي:



لتسهيل عملية قراءة البيانات من منفذ الطابعة ، اقترح عليك استخدام سلك طابعة عادي وقطعه من المنتصف والبدء بتوصيل الأسلاك ، أو شراء قطعة لمنفذ الطابعة وتقوم أنت بلحم الأسلاك يدوياً ، سيكون الأمر أسهل عليك من استخدام المنفذ مباشرة وتوصيل الأسلاك به.

البرمجة:

بعد انتهائنا من توصيل الكومبيوتر بالدائرة الكهربائية ، جاء دور البرمجة لنتمكن من التحكم بهذا الجهاز ، الطريقة الأسهل للقيام بذلك - من خلال #C مثلاً - هو عمل زر يقوم بفتح واغلاق ال Pins الموجودة بالطريقة التالية:

```
using System.Runtime.InteropServices
[DllImport("inpout32.dll", EntryPoint="Out32")]
public static extern void Output(int adress, int value);
```

ومن ثم لفتح pin معينة قم باستخدام الأمر التالي:

```
Output(888, 1);
```

لفتح الثانية قم بكتابة 2 ، الثالثة قم بكتابة 4 وهكذا ، ولفتح الثانية والثالثة مثلاً :

2 + 4 = 6.

طبعاً لا تنس ان تكون لديك مكتبة Inpout32.dll ولربما تحتاج إلى هذه المكتبة بدل الافتراضية هنا:
http://www.dll-files.com/dllindex/dll-files.shtml?inpout32

الخطوة الثانية هي البدء بتطبيق برمجة عادية جداً على هذا المنفذ ، يمكنك استخدام Timer مثلاً لفتح واغلاق المصباح ، يمكنك فتح المصباح في مواعيد معينة مسجلة مثلاً في قاعدة بيانات ، يمكن أن يكون الجهاز الكهربائي جرس وليس مصباح ويستخدم في عمل منبه ... يمكن أن يتم تشغيل جهاز كهربائي بناء على شروط معينة ، باستخدام سينسور للضوء بحيث يتم تشغيل الضوء في حالة اغلاق المصباح الرئيسي مثلاً ، او حتى باستخدام كاميرات مراقبة لتحديد اطلاق جرس ، تشغيل جهاز ... الخ ، والتطبيقات كما ترى كثيرة جداً ، وانت فقط من تستطيع تحديد البرمجة المناسبة لك.

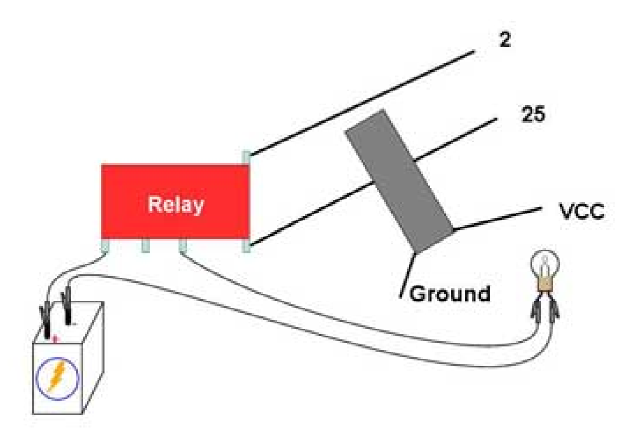
الحماية:

بدءاً من هذا الجزء سوف تجد بعض الأمور المتقدمة نسبياً ، إذا لم تستطع فهم شيء هنا فقم بتجاوزه إلى المرحلة التالية ، ولكن لا تنس أن هذا خطر.

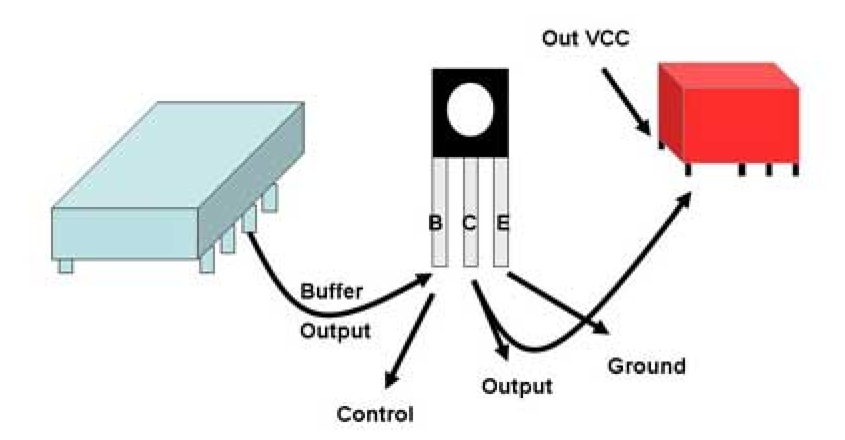
كما ترى ، فمن الخطوة بمكان الاستمرار في عمل هذه الدائرة الكهربائية ، والسبب بكل بساطة أنه في حالة حدوث أي Shortage في الدائرة الكهربائية فسوف يقوم ذلك بإعطاب ال MotherBoard مباشرة ، ومع أنها حالة ليست منتشرة إلا أننا لا بد من توفير حماية لل MotherBoard للحالات الطارئة ، هذا هو ما سنسعى إليه في هذه الخطوات.
الحل البسيط الذي سنلجأ إليه في حالتنا لحماية اللوحة الأم هو باستخدام Buffer، وهي Chip تقوم بعملية مرور للتيار الكهربائي في حالة وعدم مروره في حالة أخرى.

طريقة العمل :

سيتم وصل الإشارة القادمة من منفذ الطابعة بال Buffer من ناحية وقرائتها من الناحية الأخرى مع ادخال اشارة 1 دوماً ليمر التيار الكهربائي ، الهدف من هذه العملية هو أنه في حالة حدوث أي مشكلة في التيار الكهربائي فإن ال Bufferوحده من سيتعطل ، وهو رخيص بالمناسبة ويعمل عمل الفيوز في دائرتنا الكهربائية هذه ، سيكون بالشكل التالي:



لكننا بهذه الطريقة سنواجه مشكلة ، والسبب أن التيار الكهربائي الخارج من ال Bufferإلى دائرتنا الكهربائية سيكون غير قادر على تشغيل ال Relay ، لذا سنعتمد على وجود ترانزستور كوسيط لادخال تيار كهربائي مختلف إلى ال Relay، بحيث يتم ادخال التيار القادم من منفذ الطابعة بداية إلى Buffer ثم إلى الترانزستور ، والذي سيتحكم في دخول مصدر كهربائي خارجي (9 فولت مثلاً) إلى ال Relay الخاص بنا ، هذا المخطط على سبيل المثال:



التطوير:

كما رأيت خلال الدرس السابق ، استخدمنا أدوات بسيطة لعمل دائرة كهربائية يتم التحكم فيها من خلال الكومبيوتر ، ولكن كما ترى فهي مبنية بطريقة سريعة وقمنا بعمل دائرة حماية خاصة لها مما أدى إلى بعض التعقيد فيها ، لذا يمكننا في التطبيقات الجدية الاستغناء عن هذه الطريقة بشراء ما يعرف باسم Parallel Port Interfaceكالذي في الصورة التالية:



طبعاً سيكون أغلى وحسب امكانياته بالطبع لكن سعر العادي منه يصل إلى حوالي 250 جنيهاً مصرياً ، طريقة عمل الدائرة بسيطة جداً ، وهي قامت بالفعل بما نقوم نحن بعمله حيث نجد Relays اضافة إلى دائرة حماية خاصة ، تستطيع هذه الدائرة تشغيل 8 أجهزة أو حسب نوعها ، يتم توصيلها بسلك عادي إلى منفذ الطابعة حيث نجد أن كل relay من الثمانية يتم التحكم به من خلال Pin في منفذ الطابعة ، بالطبع كل ما عليك هو توصيل أجهزتك الكهربائية بهذا ال Relayليتولى هو عنك الباقي حيث تقوم بتشغيله أو برمجته كما تعلمنا في القسم السابق من هذا الدرس.

هذا هو واحد من هذه الأنواع:
http://www.kemo-electronic.de/en/bausaetze/b210/index.htm

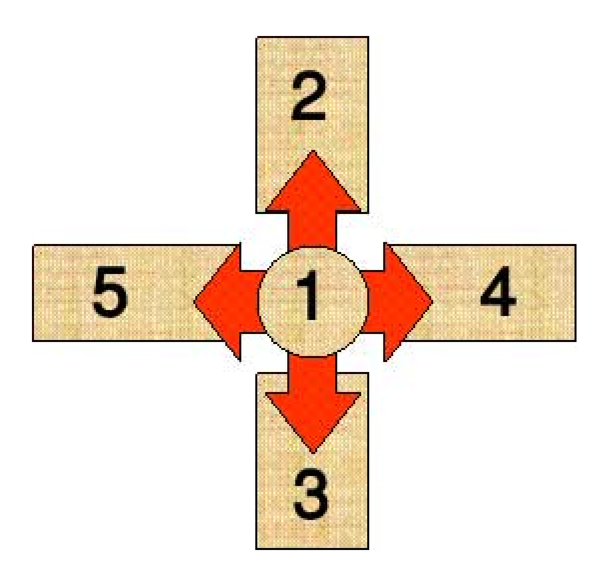
الدرس الحادي عشر : قيادة عربة ريموت كنترول من خلال الكومبيوتر

الأدوات التي نحتاجها لهذا الدرس:

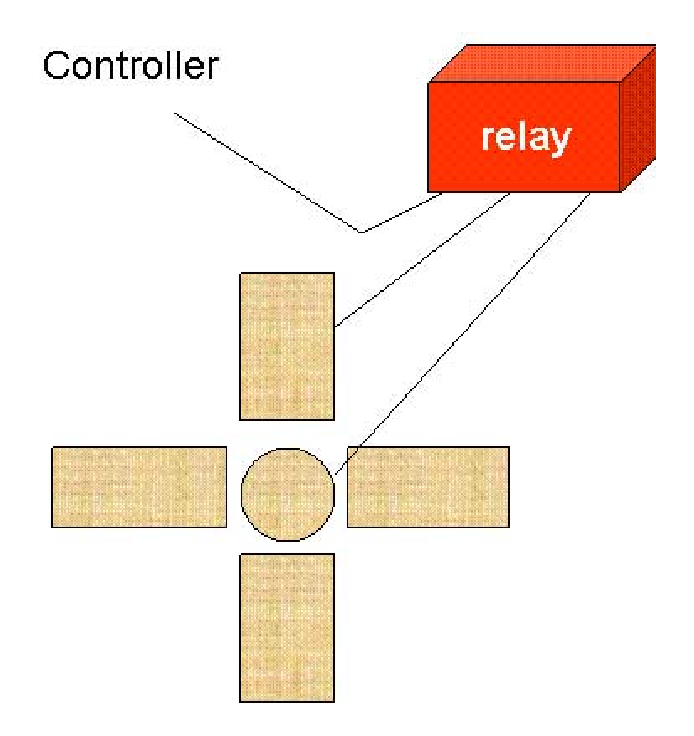
1- سيارة بسيطة (لعبة) تعمل بالريموت كنترول.
2- Relay .
3- Transisitor .
4- Buffer .
5- مجموعة من الأسلاك والتوصيلات.

والآن وبعد متابعتك للدروس السابقة ، فأنت تدرك تماماً أن بامكانك التحكم في عبور التيار بين النقاط المختلفة عن طريق الكومبيوتر ، هذا هو المبدأ الذي سنعتمد عليه في درسنا الحالي.

سنبدأ الآن بفك جهاز الريموت كنترول الخاص بالعربة ، الريموت في ابسط صوره يكون بالشكل التالي:



إذا لامس الطرف الأول الطرف رقم (2 (تسير العربة للأمام ، (3) للخلف ، (4) و (5) لليمين واليسار ، لذا سنجرب وضع Relay على التوصيلة للامام مثلاً بالشكل التالي:



يمكنك ان تلاحظ بكل سهولة امكانية التحكم في ال relay من خلال الكومبيوتر وهو ما يعني بالتالي تحريك العربة عن طريق الكومبيوتر ، لذا سنضع ثلاث ريليهات اخرى لباقي الاتجاهات.

لا تنس اضافة دائرة للحماية كتلك التي شرحناها في الدرس السابق ، ايضاً يمكنك استخدام parallel port interface مباشرة ، لنفترض اننا قمنا بالتوصيل على ال pins من 1 إلى 4.

البرمجة:

بداية سنقوم برسم فورم بسيط يحتوي على اربعة ازرار ، سنقوم بداية باستدعاء مكتبة inpout32.dll كما تعلمنا في الدروس السابقة بالشكل التالي :

```
using System.Runtime.InteropServices
[DllImport("inpout32.dll", EntryPoint="Out32")]
public static extern void Output(int adress, int value);
```

في زر التحرك للامام سنكتب الكود التالي:

```
Output(888, 1);
```

في باقي الاتجاهات سوف نقوم فقط باستبدال رقم 1 بالارقام 2 و 4 و 8 كما تعلمنا سابقاً.

الريموت السابق كان في ابسط صوره ، بعض الريموتات اكثر تعقيداً ، بل ان بعض الريموتات تحتوي على تحكمات اكثر مثل الفرامل ، تشغيل الانوار .. الخ ، وكل ذلك يمكن التحكم فيه عن طريق الكومبيوتر.

قبل النهاية ، يمكنك اضافة لمسة جمالية بسيطة عن طريق تحريك العربة عن طريق اسهم الكيبورد بدلاً من الضغط عليها بالماوس ، يتم اقتناص احداث الماوس في key Down فقط حيث لا يمكن قراءتها في key press.

لذا قم بتعديل الخاصية KeyPreview للفورم إلى True ، ومن ثم اكتب في حدث Form_KeyDown الأمر التالي:

```
if (e.KeyValue.ToString() == "37")
// left
else if (e.KeyValue.ToString() == "38")
// up
else if (e.KeyValue.ToString() == "39")
// right
else if (e.KeyValue.ToString() == "40")
// down
```

الدرس الأخير : مقدمة إلى عالم الروبوتات

في درسنا الأخير في دورة منفذ الطابعة ، سنطل بطريقة بسيطة على عالم الروبوتات نظراً لوجود صلة بينه وبين عالم ال Physical Computing ، وقبل البداية نود ان نعرف الروبوت.

يقصد بالروبوت اي شيء يتمتع بقدر معين القدرات المشابهه للانسان والتي تقوم بمهام بديلة عنه سواء اكان حقيقياً ام روبوت وهمي على شكل برنامج كومبيوتر ، ومن هذا نستطيع ان نعرف ان محرك البحث هو روبوت ، العربة التي دخلت إلى الاهرامات للاسكتشاف او صعدت للقمر للاستكشاف هي روبوت ايضاً.

لكن ما يتبادر إلى ذهن الناس دائماً عند اطلاق كلمة روبوت هو ذلك الكائن الشديد الشبه بالانسان الذي يملك قدمين ويدين لتنفيذ بعض المهام وخلافه ، إلا أن هذه النظرة قاصرة نوعاً ما ، فهناك انواع شديدة التعقيد من الروبوتات التي ربما لا تشابه الإنسان في اي شيء.

في درسنا اليوم سوف نتحدث عن النوع الحقيقي من الروبوتات وليس عن برامج الكومبيوتر ، وسنبدأ بتعريف : ما هي المكونات الاساسية لاي روبوت ، ومتى نستطيع ان نطلق عليه هذا الاسم "روبوت" ؟

1- جسم : body structure يحدد الشكل بنسبة كبيرة طبيعة المهمة التي يقوم بها الروبوت وتساهم بنسبة كبيرة في نجاحه ، لذا يعد الشكل عنصراً اساسياً ليتمكن الروبوت من العمل (روبوت تقطيع الاحجار مثلاً يحتاج إلى بنية صلبة ، اذرع من الفولاذ منتهية بأدوات التقطيع مثلاً.

2- عضلات : muscle تقوم بالتحكم في حركاته ، في الغالب تكون عضلات الروبوت هي مجموعة من التروس والمواتير التي تتحكم في جميع حركاته.

3- حساسات : sensor system تقوم بالتحكم بالمدخلات إليه ، مثلاً قد نحتاج إلى كاميرا مراقبة تقوم مقام العين ، مايكروفون لالتقاط الاصوات ، قد نحتاج إلى حساسات للمياة او للحرارة او الرطوبة او خلافه حسب حاجة الروبوت.

4- عقل : brain وهو الذي يتحكم في تفسير وفهم المدخلات المختلفة ومن ثم تنفيذ الحركات عن طريق العضلات او وسيلة الاخراج والتنفيذ الخاصة بالروبوت ، يكون عادة على شكل برنامج كومبيوتر على مايكروكنترولر خاص بالروبوت.

4- مصدر طاقة : power supply ايضاً يحتاج الروبوت لمصدر لطاقته ، في العادة يتم استخدام بطاريات مخصصة حسب حجم واحتياجات الروبوت.

مثال سريع:

سنحاول الآن شرح مثال بسيط على روبوت ، وسنحاول تطبيق مبادئ الروبوتات الخمسة عليه.

سيقوم الروبوت الخاص بنا بالبحث عن الأوراق الزرقاء مثلاً في وسط مجموعة من الاوراق ، وعندما يعثر على واحدة يقوم بطباعة كلمة بسيطة عليها.

المكونات التي نحتاج إليها:

-على صعيد الجسم : نحتاج إلى جسم قابل للحركة - ذراع تحمل ختماً ما - واخيراً حامل قابل للحركة في كل الاتجاهات ليقوم بحمل الكاميرا.
-على صعيد العضلات : نحتاج إلى موتور لتحريك الجسم (بالعجلات مثلاً) - موتور اخر للتحكم في الذراع التي تحمل الختم ، واخيراً موتور ثالث لتحريك حامل الكاميرا.
-على صعيد وسائل الاحساس : نحتاج إلى كاميرا لتقوم بالتقاط الصور.
-على صعيد مصدر الطاقة : سنكتفي ببطارية للكاميرا وأخرى للعربة المتحركة التي تحملها ، بطارية أخيرة للموتور الخاص بالأذرعة والحامل.
-وأخيراً على صعيد العقل : نحتاج إلى برنامج كومبيوتر لتحريك الكاميرا ، تحريك الذراع ، والجوريزم معالجة الصور والبحث عن اللون الأزرق.

العمل:

-قم بشراء عربة عادية بالبطاريات ، سيكون هذا هو الجسم الخاص بنا والذي سنقوم بتحريكه عن طريق الكومبيوتر كما تعلمنا سابقاً.
-قم بشراء كاميرا بسيطة وقم بتوصيلها بالكومبيوتر ، قم بوضعها على حامل وربطها بموتور عادي وربط الموتور بالكومبيوتر ايضاً لتحريكه من اجل التحرك في جميع الاتجاهات.
-ختم بسيط من اي مكتبة ، اربطه بحامل وقم بتحويل حركة الموتور الدائرية إلى افقية باستخدام التروس.
-الآن في برنامجك قم باعطاء الأوامر للحامل بالتحرك بالكاميرا في جميع الاتجاهات كل ثانية ، والتقاط صورة ، قم باستخدام احد الجوريزمات قراءة الصور والبحث عن وجود اللون الازرق.
-عندما تجد اللون الازرق قم بحساب زاوية انحراف الكاميرا ، قم بالتحرك بالعربة في هذا الاتجاه حتى تصل إليها . قم بتحريك الختم لاعلى واسفل.

تطويرات:

-سيكون من المربك جداً عمل كل هذا الجهد باستخدام توصيلات واسلاك ، يمكن استخدام وحدات ارسال واستقبال لاسكلي.
-يمكن ايضاً الاستغناء على الكومبيوتر ووضع microcontrolller بدلاً منه في العربة.

مسابقة روبوكون:

هي مسابقة عالمية تقام فعالياتها سنوياً بين الطلاب من الجامعات المختلفة لبناء روبوت يقوم بتنفيذ مهمة محددة ، توجد تصفيات لها في مصر ويتم سفر الفرق الفائزة إلى المسابقة العالمية.

لا اعرف إذا كانت هناك تصفيات في دول عربية اخرى ، ولكن هذا الرابط لتوقيتات المسابقة لهذا العام:
http://www.egyptradio.tv/robocon/table.htm

مسابقة الروبوكون للعام الفائت كانت بخصوص عمل روبوت يقوم بالبحث عن بعض الحلقات ووضعها داخل عمود ، المطلوب من المسابقة موجود على شكل فيديو على الرابط التالي:
http://www. youtube.com/watch?v=t15r9-ZXFwM&feature=related

اما مسابقة هذه السنة والتي ستقام نهائياتها في الهند ، حيث ستقوم الروبوتات بنقل الاطعمة وسرقتها احياناً ، تجد في الرابط الأول شروط وطبيعة المسابقة وكافة التفاصيل عنها إذا احببت الاشتراك فيها لهذه السنة.

اختم هذا الدرس ببعض الفيديوهات:

روبوتات كرة القدم الأمريكية ☺
http://www. youtube.com/watch?v=O5KaBtWwu7I

روبوت اللمونة الدايخة ☺
http://www. youtube.com/watch?v=YaoyHHMRXQ4

المصارعة الحرة العالمية ☺
http://www. youtube.com/watch?v=eyorb9MHYr0&NR=1

في نهاية هذه الدورة اسأل الله ان اكون قد قدمت بعض الفائدة ، فما كان فيها من صواب فمن الله وما كان فيها من خطأ فمن نفسي والشيطان.

والله الموفق...
والسلام عليكم ورحمة الله وبركاته.